



CCCCCCCC	HH	HH	KK	KK	HH	HH	DDDDDDDD	RRRRRRRR		
CCCCCCCC	HH	HH	KK	KK	HH	HH	DDDDDDDD	RRRRRRRR		
CC	HH	HH	KK	KK	HH	HH	DD	DD	RR	RR
CC	HH	HH	KK	KK	HH	HH	DD	DD	RR	RR
CC	HH	HH	KK	KK	HH	HH	DD	DD	RR	RR
CC	HH	HH	KK	KK	HH	HH	DD	DD	RR	RR
CC	HH	HH	KK	KK	HH	HH	DD	DD	RR	RR
CC	HH	HH	KK	KK	HH	HH	DD	DD	RR	RR
CC	HH	HH	KK	KK	HH	HH	DD	DD	RR	RR
CC	HH	HH	KK	KK	HH	HH	DD	DD	RR	RR
CCCCCCCC	HH	HH	KK	KK	HH	HH	DDDDDDDD	RRRRRRRR	RR	RR
CCCCCCCC	HH	HH	KK	KK	HH	HH	DDDDDDDD	RRRRRRRR	RR	RR
										....
										....
										....
										....

LL	IIIIII	SSSSSSSS
LL	IIIIII	SSSSSSSS
LL	II	SS
LL	II	SS
LL	II	SS
LL	II	SS
LL	II	SSSSSS
LL	II	SSSSSS
LL	II	SS
LL	II	SS
LL	II	SS
LL	II	SS
LLLLLLLLLL	IIIIII	SSSSSSSS
LLLLLLLLLL	IIIIII	SSSSSSSS

.....



1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57

```

0001 0 MODULE CHKHDR (
0002 0
0003 0     LANGUAGE (BLISS32),
0004 0     IDENT = 'V04-000'
0005 1 BEGIN
0006 1
0007 1 |
0008 1 |*****|
0009 1 |*|
0010 1 |*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY |*
0011 1 |*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS. |*
0012 1 |*  ALL RIGHTS RESERVED. |*
0013 1 |*|
0014 1 |*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED |*
0015 1 |*  ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE |*
0016 1 |*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER |*
0017 1 |*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY |*
0018 1 |*  OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY |*
0019 1 |*  TRANSFERRED. |*
0020 1 |*|
0021 1 |*  THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE |*
0022 1 |*  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT |*
0023 1 |*  CORPORATION. |*
0024 1 |*|
0025 1 |*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS |*
0026 1 |*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL. |*
0027 1 |*|
0028 1 |*|
0029 1 |*****|
0030 1 |
0031 1 |++
0032 1 |
0033 1 | FACILITY: F11ACP Structure Level 1
0034 1 |
0035 1 | ABSTRACT:
0036 1 |
0037 1 |   This routine verifies that the block given it is in fact a
0038 1 |   file header. If file number and/or file sequence number are also
0039 1 |   supplied, they are checked as well.
0040 1 | ENVIRONMENT:
0041 1 |
0042 1 |   STARLET operating system, including privileged system services
0043 1 |   and internal exec routines.
0044 1 |
0045 1 | --
0046 1 |
0047 1 |
0048 1 | AUTHOR: Andrew C. Goldstein, CREATION DATE: 13-Dec-1976 16:11
0049 1 |
0050 1 | MODIFIED BY:
0051 1 |
0052 1 |   A0101   ACG0146           Andrew C. Goldstein,   22-Feb-1980 21:42
0053 1 |           Change file sequence number check to no such file
0054 1 |
0055 1 |   A0100   ACG00001        Andrew C. Goldstein, 10-Oct-1978 20:02
0056 1 |           Previous revision history moved to F11A.REV
0057 1 |**

```

CHKHDR  
V04-000

K 7  
16-Sep-1984 00:49:10  
14-Sep-1984 12:29:19

VAX-11 Bliss-32 V4.0-742  
DISK\$VMSMASTER:[F11A.SRC]CHKHDR.B32;1 Page (1) 2

```
.. 58      0058 1
.. 59      0059 1
.. 60      0060 1 LIBRARY 'SYSSLIBRARY:LIB.L32';
.. 61      0061 1 REQUIRE 'SRCS:FCPDEF.B32';
```

CHK  
V04



```

63 0376 1 GLOBAL ROUTINE CHECK_HEADER (HEADER, FILE_ID) =
64 0377 1
65 0378 1 |++
66 0379 1
67 0380 1 FUNCTIONAL DESCRIPTION:
68 0381 1
69 0382 1 This routine verifies that the block given it is in fact a
70 0383 1 file header. If file number and/or file sequence number are also
71 0384 1 supplied, they are checked as well.
72 0385 1
73 0386 1 CALLING SEQUENCE:
74 0387 1 CHECK_HEADER (ARG1, ARG2)
75 0388 1
76 0389 1 INPUT PARAMETERS:
77 0390 1 ARG1: address of header image
78 0391 1 ARG2: address of file ID
79 0392 1
80 0393 1 IMPLICIT INPUTS:
81 0394 1 NONE
82 0395 1
83 0396 1 OUTPUT PARAMETERS:
84 0397 1 NONE
85 0398 1
86 0399 1 IMPLICIT OUTPUTS:
87 0400 1 USER_STATUS contains code if not valid
88 0401 1
89 0402 1 ROUTINE VALUE:
90 0403 1 0 if garbage
91 0404 1 1 if valid and correct file header
92 0405 1 2 if deleted file header
93 0406 1
94 0407 1 SIDE EFFECTS:
95 0408 1 NONE
96 0409 1
97 0410 1 |--
98 0411 1
99 0412 2 BEGIN
100 0413 2
101 0414 2 MAP
102 0415 2 HEADER : REF BBLOCK, ! file header arg
103 0416 2 FILE_ID : REF BBLOCK; ! file ID arg
104 0417 2
105 0418 2 LOCAL
106 0419 2 MAP_AREA : REF BBLOCK; ! pointer to header map area
107 0420 2
108 0421 2 EXTERNAL ROUTINE
109 0422 2 CHECKSUM; ! compute file header checksum
110 0423 2
111 0424 2
112 0425 2 ! First check the structure level.
113 0426 2 !
114 0427 2
115 0428 2 IF .HEADER[FH1$W_STRUCLEV] NEQ FH1$C_LEVEL1
116 0429 2 THEN (ERR_STATUS=(SS$_FILESTRUCT); RETURN 0);
117 0430 2
118 0431 2 ! Now point to the map area and make sure that the extension
119 0432 2 ! RVN is zero (no multi-volume supported yet.)

```

```
: 120 0433 2 ! Also check the retrieval pointer format data.
: 121 0434 2 !
: 122 0435 2 !
: 123 0436 2 MAP_AREA = .HEADER + .HEADER[FH1$B_MPOFFSET]*2;
: 124 0437 2 !
: 125 0438 2 IF .MAP_AREA[FM1$B_EX_RVN] NEQ 0
: 126 0439 2 OR .MAP_AREA[FM1$B_COUNTSIZE] NEQ 1
: 127 0440 2 OR .MAP_AREA[FM1$B_LBNSIZE] NEQ 3
: 128 0441 2 THEN (ERR_STATUS (SS$_FILESTRUCT); RETURN 0);
: 129 0442 2 !
: 130 0443 2 ! Check the retrieval pointer counts for consistency with the
: 131 0444 2 ! available space.
: 132 0445 2 !
: 133 0446 2 !
: 134 0447 2 IF .MAP_AREA[FM1$B_INUSE] GTRU .MAP_AREA[FM1$B_AVAIL]
: 135 0448 2 OR .MAP_AREA[FM1$B_AVAIL] GTRU
: 136 0449 2 255 - (.MAP_AREA + FM1$C_POINTERS - .HEADER) / 2
: 137 0450 2 THEN (ERR_STATUS (SS$_BADFILEHDR); RETURN 0);
: 138 0451 2 !
: 139 0452 2 ! At this point, we have verified that the block at least once was a
: 140 0453 2 ! valid file header.
: 141 0454 2 !
: 142 0455 2 ! Look at the file number in the header. If zero, this is a
: 143 0456 2 ! deleted header.
: 144 0457 2 !
: 145 0458 2 !
: 146 0459 2 IF .HEADER[FH1$W_FID_NUM] EQL 0
: 147 0460 2 THEN (ERR_STATUS (SS$_NOSUCHFILE); RETURN 2);
: 148 0461 2 !
: 149 0462 2 ! Now compute the header checksum.
: 150 0463 2 !
: 151 0464 2 !
: 152 0465 2 IF NOT CHECKSUM (.HEADER)
: 153 0466 2 THEN (ERR_STATUS (SS$_BADCHKSUM); RETURN 2);
: 154 0467 2 !
: 155 0468 2 ! Check file number and file sequence number.
: 156 0469 2 !
: 157 0470 2 !
: 158 0471 2 IF .HEADER[FH1$W_FID_NUM] NEQ .FILE_ID[FID$W_NUM]
: 159 0472 2 THEN (ERR_STATUS (SS$_FILENUMCHK); RETURN 2);
: 160 0473 2 !
: 161 0474 2 IF .HEADER[FH1$W_FID_SEQ] NEQ .FILE_ID[FID$W_SEQ]
: 162 0475 2 THEN (ERR_STATUS (SS$_NOSUCHFILE); RETURN 2);
: 163 0476 2 !
: 164 0477 2 ! Header is ok.
: 165 0478 2 !
: 166 0479 2 !
: 167 0480 2 RETURN 1;
: 168 0481 2 !
: 169 0482 1 END; ! end of routine CHECK_HEADER
```

```
.TITLE CHKHDR
.IDENT \V04-000\
.EXTRN CHECKSUM, USER_STATUS
```



				.PSECT	\$CODE\$,NOWRT,2	
			000C 00000	.ENTRY	CHECK HEADER, Save R2,R3	: 0376
	53	00000000G	00 9E 00002	MOVAB	USER STATUS, R3	: 0428
	52	04	AC D0 00009	MOVL	HEADER, R2	: 0436
0101	8F	06	A2 B1 0000D	CMPW	6(R2), #257	: 0438
			19 12 00013	BNEQ	1\$	: 0439
	50	01	A2 9A 00015	MOVZBL	1(R2), R0	: 0441
	50		6240 3E 00019	MOVAW	(R2)[R0], MAP_AREA	: 0447
		01	A0 95 0001D	TSTB	1(MAP_AREA)	: 0449
			0C 12 00020	BNEQ	1\$	: 0450
	01	06	A0 91 00022	CMPB	6(MAP_AREA), #1	: 0459
			06 12 00026	BNEQ	1\$	: 0465
	03	07	A0 91 00028	CMPB	7(MAP_AREA), #3	: 0466
			0A 13 0002C	BEQL	2\$	: 0471
	73		63 E9 0002E 1\$:	BLBC	USER STATUS, 10\$	: 0472
	63	08C0	8F B0 00031	MOVW	#2240, USER_STATUS	: 0474
			6C 11 00036	BRB	10\$	: 0475
	09	A0	A0 91 00038 2\$:	CMPB	8(MAP_AREA), 9(MAP_AREA)	: 0480
			17 1A 0003D	BGTRU	3\$	: 0482
51			50 C3 0003F	SUBL3	MAP_AREA, R2, R1	: 0482
			0A C2 00043	SUBL2	#10, R1	: 0482
			02 C6 00046	DIVL2	#2, R1	: 0482
			C1 9E 00049	MOVAB	255(R1), R1	: 0482
51	09	A0	00FF 08	CMPZV	#0, #8, 9(MAP_AREA), R1	: 0482
			0A 1B 00054	BLEQU	4\$	: 0482
	4B		63 E9 00056 3\$:	BLBC	USER STATUS, 10\$	: 0482
	63	0810	8F B0 00059	MOVW	#2064, USER_STATUS	: 0482
			44 11 0005E	BRB	10\$	: 0482
		02	A2 B5 00060 4\$:	TSTW	2(R2)	: 0482
			2F 13 00063	BEQL	7\$	: 0482
			52 DD 00065	PUSHL	R2	: 0482
0000G	CF		01 FB 00067	CALLS	#1, CHECKSUM	: 0482
	0A		50 E8 0006C	BLBS	R0, 5\$	: 0482
	2A		63 E9 0006F	BLBC	USER STATUS, 8\$	: 0482
	63	0808	8F B0 00072	MOVW	#2056, USER_STATUS	: 0482
			23 11 00077	BRB	8\$	: 0482
	50	08	AC D0 00079 5\$:	MOVL	FILE ID, R0	: 0482
	60	02	A2 B1 0007D	CMPW	2(R2), (R0)	: 0482
			0A 13 00081	BEQL	6\$	: 0482
	16		63 E9 00083	BLBC	USER STATUS, 8\$	: 0482
	63	08B0	8F B0 00086	MOVW	#2224, USER_STATUS	: 0482
			0F 11 0008B	BRB	8\$	: 0482
	02	A0	A2 B1 0008D 6\$:	CMPW	4(R2), 2(R0)	: 0482
			0C 13 00092	BEQL	9\$	: 0482
	05		63 E9 00094 7\$:	BLBC	USER STATUS, 8\$	: 0482
	63	0910	8F B0 00097	MOVW	#2320, USER_STATUS	: 0482
			50 02 D0 0009C 8\$:	MOVL	#2, R0	: 0482
			04 0009F	RET		: 0482
	50		01 D0 000A0 9\$:	MOVL	#1, R0	: 0482
			04 000A3	RET		: 0482
			50 D4 000A4 10\$:	CLRL	R0	: 0482
			04 000A6	RET		: 0482

; Routine Size: 167 bytes, Routine Base: \$CODE\$ + 0000



CHKHDR  
V04-000

B 8  
16-Sep-1984 00:49:10  
14-Sep-1984 12:29:19

VAX-11 Bliss-32 V4.0-742  
DISK\$VMMASTER:[F11A.SRC]CHKHDR.B32;1 Page 6 (2)

: 170 0483 1  
: 171 0484 1 END  
: 172 0485 0 ELUDOM

PSECT SUMMARY

Name Bytes Attributes  
\$CODE\$ 167 NOVEC,NOWRT, RD , EXE,NOSHR, LCL, REL, CON,NOPIE,ALIGN(2)

Library Statistics

File	----- Total	Symbols Loaded	----- Percent	Pages Mapped	Processing Time
_\$255\$DUA28:[SYSLIB]LIB.L32;1	18619	19	0	1000	00:02.0

COMMAND QUALIFIERS

BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LISS:CHKHDR/OBJ=OBJ\$:CHKHDR MSRC\$:CHKHDR/UPDATE=(ENH\$:CHKHDR)

: Size: 167 code + 0 data bytes  
: Run Time: 00:07.9  
: Elapsed Time: 00:23.7  
: Lines/CPU Min: 3697  
: Lexemes/CPU-Min: 13921  
: Memory Used: 108 pages  
: Compilation Complete



FCPDEF B32	ACPCNTR LIS	CHKSUM LIS	CHKPRO LIS	DEACCS LIS
BADSEN LIS	CLEUP LIS	CPYAM LIS	CHKHDR LIS	COMMON LIS
CREHDR LIS	CREWIN LIS	ACCESS LIS	ALLOB LIS	CHKDMD LIS
DELETE LIS	CREATE LIS	CREFCB LIS		